

Amendments to the Claims:

This listing of claims will replace all prior versions, and listing, of claims in the application:

Listing of the Claims:

1. (Canceled)

2. (Currently Amended) A ~~method~~ recited in claim 1, wherein said method further comprises computing environment as recited in claim 23, wherein said file-system-independent component is further operable to:

denying deny said at least one operation when said determining determines that said at least one operation should not be allowed.

3-6. (Canceled)

7. (Currently Amended) A computing environment as recited in claim 23 ~~method~~ recited in claim 4, wherein said at least one mandatory lock category can be a Byte-Range lock or a Shared Resource lock.

8. (Currently Amended) A ~~method~~ computing environment as recited in claim 7, wherein the type of said Byte-Range lock can be exclusive or shared.

9. (Currently Amended) A ~~method~~ computing environment as recited in claim 7, wherein said Shared Resource lock can have a deny mode associated with it;

and

wherein said deny mode can be defined with respect to reading or writing of said file.

10. (Currently Amended) A ~~method~~ computing environment as recited in claim 23 [[1]], wherein said at least one operation can be a read, write, delete, rename, memory map or change size operation.

11. (Currently Amended) A computing environment as recited in claim 23, wherein said file-system-independent component is operable to method of processing requests to perform operations on a file which is capable of being stored in a UNIX file system, said method comprising:

(a) ~~receiving~~ receive a request to perform an operation on a ~~UNIX~~ file which has a mandatory Byte-Range lock associated with it;

(b) ~~determining~~ determine whether said requested operation may affect a byte range of the ~~UNIX~~ file; said byte range representing a portion of said file which is associated with the mandatory Byte-Range lock; and

(c) ~~determining~~ determine whether said operation is compatible with said Byte-Range lock when said determining (b) determines that said requested operation may affect said byte range.

12. (Currently Amended) A ~~method~~ computing environment as recited in claim 11, wherein said file-system-independent component is further operable to method further comprises:

~~determining~~ determine whether said request was made by the owner of said Byte-Range lock when said determining (c) determines that said operation is not compatible with said Byte-Range.

13. (Currently Amended) A ~~method~~ computing environment as recited in claim 11, wherein said mandatory Byte-Range lock can be an exclusive or shared lock.

14-18. (Canceled)

19. (Currently Amended) A computing environment as recited in claim 23, wherein said file-system-independent operating system is further operable to method of processing a request to change the size of a file which is capable of being stored in a UNIX file system; said method comprising:

~~determining~~ determine whether a mandatory Byte-Range lock or a mandatory Shared Resource lock is associated with said file;

~~determining~~ determine whether said Shared Resource lock includes a deny write operation when said ~~determining~~ determines it is determined that a mandatory Shared Resource lock is associated with said file; and

~~identifying~~ identify a region of said file which ~~may~~ would be affected by said request to change changing the size of said file when ~~said determining~~ determines it is determined that a mandatory Byte-Range lock has been associated with the file.

20. (Currently Amended) A ~~method~~ computing environment as recited in claim 19, wherein said ~~method~~ file-system-independent component is further operable to comprises:

~~determining~~ determine whether said identified region intersects a locked region of said file; and

~~allowing~~ allow said request to change said file size when ~~said determining~~ determines it is determined that said identified region does not intersect with said locked region of said file.

21. (Currently Amended) A ~~method~~ computing environment as recited in claim 19, wherein said ~~method~~ file-system-independent component is further operable to comprises:

~~determining~~ determine whether said request was made by the owner of said mandatory Byte-Range lock or mandatory Shared Resource lock.

22. (Currently Amended) A ~~method~~ computing environment as recited in claim 19, wherein said request to change said file size is allowed when said determining whether said Shared Resource lock does not include a deny write operation.

23. (Currently Amended) A ~~UNIX~~ computing environment, comprising:

a ~~UNIX~~ file system capable of storing one or more files therein, wherein said file system does not provide mandatory locks for files stored in said first file system:

an operating system that can access files stored in said file system;

a file-system-independent component of said operating system, wherein said - system-independent component effectively implements mandatory locks for said file system and is operable to:

receive a request to perform at least one operation on a file stored in said file system;

determine whether at least one mandatory lock is associated with said file;

determine a mandatory lock category or type for said at least one mandatory lock when said determining determines that at least one mandatory lock is associated with said file;

determine whether said least one mandatory lock category or type is compatible with said at least one operation;

determine at least partly based on compatibility of said lock category or type with said at least one operation whether said at least one operation should be allowed; and

allow said at least one operation when said determining determines that said at least one operation should be allowed; and

wherein said at least one operation can be changing the size of a file in said file-system.

~~wherein said UNIX computing environment is capable of enforcing mandatory locks for said one or more files in said UNIX file system; and~~

~~wherein said mandatory locks comprise of at least two mandatory lock categories.~~

24. (Canceled)

25. (Currently Amended) A ~~UNIX~~ computing environment as recited in claim 23 24, wherein said at least two categories comprise Byte-Range locks and Shared Resource locks.

26. (Currently Amended) A ~~UNIX~~ computing environment as recited in claim 23, wherein said mandatory locks can be enforced with respect to read, write, delete, rename, memory map, or change size operations.

27-29. (Canceled)

30. (New) A computer readable medium that stores computer program code for said file-system-independent component recited in claim 23.